CLAIMS

5

  1. A system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action including an action method table comprising a

10 plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the system further having a run-time information library which includes a plurality of information objects, each information object

15 representing one of an action and a container type.

  2. The system of claim 1 wherein each information object contains information about the represented action or container type including members of a group consisting of a globally unique identifier, a type name, a

20 system type, a pointer to ancestor type information, a table information pointer for any derived types, a pointer to a class factory which instantiates the type, and combinations thereof.

  3. The system of claim 1 wherein each information object

25 representing an action includes information necessary to instantiate the action at run-time.

  4. The system of claim 1 wherein each information object representing a container type includes information necessary to instantiate a

30 container based on the container type at run-time.

5. The system of claim 1 wherein each information object representing a container type includes information necessary to expand and fill in the action method table of each action at run-time such that a container

5   instantiated based on the container type may be handled when encountered by each action when traversing the directed graph.

6. A method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a

10 plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select

15 the action method to be executed on the encountered container, the method for instantiating an action during run-time, the instantiating action depending from a parent action, the method comprising:

querying a run-time information library for information on the instantiating action, the information including the parent action of the instantiating

20 action and the specific action methods associated with the instantiating action; and

instantiating the action based on such information by:

ensuring that the parent action to the instantiating

action is instantiated;

25 copying the action method table of the parent action and employing the copied action method table as the action method table of the instantiating action; and

updating appropriate ones of the action methods in the copied method table to be the specific action methods associated with the

30 instantiating action based on the information obtained from the run-time

information library, wherein at least some of the action methods in the action method table of the instantiating action are not updated and thus are action methods inherited from the parent action.

7.   The method of claim 6 wherein the system further has a global table including the action methods associated with the instantiated actions, and wherein the action method table of each action comprises a plurality of pointers pointing to the action methods in the global table, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the pointer to the action method in the global table to be executed on the encountered container, the method comprising:

copying the specific action methods associated with the instantiating action to the global table; and

updating appropriate ones of the pointers in the copied action method table to point at the specific action methods associated with the instantiating action based on the information obtained from the run-time information library, wherein at least some of the pointers in the action method table of the instantiating action are not updated and thus are pointed at action methods inherited from the parent action.

8.   The method of claim 6 wherein the system further has a run-time action table maintaining a list of all actions instantiated in the system, the method further comprising first determining from the run-time action table that the action is not instantiated.

9.   The method of claim 6 wherein the actions in the system are organized into an action tree which branches out from a generalized base action to a plurality of more specific actions at multiple levels of specificity, each non-base action directly depending from a parent action based on the action tree, and

wherein ensuring that the parent action to the instantiating action is instantiated comprises going back through the action tree from the instantiating action to the base action as far back as is necessary to ensure that the parent action to the instantiating action is instantiated.

5

   10.   A method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the method for developing a new action to be instantiated during run-time, the new action for imparting a particular functionality, the method comprising:

   defining a parent action for the new action to depend from based on the particular functionality to be imparted;

   defining specific action methods to be performed by the new action with regard to at least some of the container types in the system based on the functionality to be imparted and thereby at least implicitly designating action methods to be inherited by the new action from the parent action, the inherited action methods to be performed by the new action with regard to the other container types in the system; and

   storing the new action including the defined parent action and defined specific action methods in a form to be retrieved by the system during run-time.

   11.   The method of claim 10 wherein storing the new action comprises storing the new action as a class available to a class factory.

30

12.   A method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated

5   including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the method for expanding and filling in the action method table of an action ('action of interest')

10   during run-time upon determining that at least one container type has been added to the system, the action of interest depending from a parent action, the method comprising:

determining how many new container types have been added;

15   expanding the action method table of the action of interest a corresponding number of entries;

querying a run-time information library for information on each added container type and the action of interest, the information including the parent action of the action of interest and a specific action method of each added

20   container type if any with regard to the action of interest; and

filling in the expanded entries of the action method table of the action of interest based on such information by:

ensuring that the parent action to the action of interest is expanded and filled in;

25   copying the corresponding entries from the action method table of the parent action and employing the copied entries as the corresponding entries in the action method table of the action of interest; and

updating appropriate ones of the action methods in the copied entries in the action method table to be the specific action methods

30   associated with the added container types if any based on the information

obtained from the run-time information library, wherein action methods in the copied entries of the action method table of the action of interest that are not updated are action methods inherited from the parent action.

5   13.   The method of claim 12 further comprising determining that a container type has been added to the system.

          14.   The method of claim 13 wherein determining that a container type has been added to the system comprises obtaining a count of all container

10   types in the system and determining whether the action method table of the action of interest has a proper size based on the count.

          15.   The method of claim 12 wherein the system further has a global table including the action methods associated with the instantiated actions,

15   and wherein the action method table of each action comprises a plurality of pointers pointing to the action methods in the global table, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the pointer to the action method in the global table to be executed on the encountered

20   container, the method comprising:

                    copying the specific action methods associated with the added container types if any to the global table; and

                    updating appropriate ones of the pointers in the copied entries of the action method table to point at the specific action methods

25   associated with the added container types based on the information obtained from the run-time information library, wherein at least some of the pointers in the action method table of the action of interest are not updated and thus are pointed at action methods inherited from the parent action.

16.   The method of claim 12 wherein the actions in the system are organized into an action tree which branches out from a generalized base action to a plurality of more specific actions at multiple levels of specificity, each non-base action directly depending from a parent action based on the action tree, and

5   wherein ensuring that the action method table of the parent action to the action of interest is expanded and filled in comprises going back through the action tree from the action of interest to the base action as far back as is necessary to ensure that the action method table of the parent action to the action of interest is expanded and filled in.

10

17.   A method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated

15   including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the method for developing a new container type to be added to the system during run-time, the

20   new container type for imparting a particular structure, the method comprising:

defining structural behavior to be imparted based on a container instantiated from the new container type;

defining specific action methods to be performed by at least some existing actions with regard to the container instantiated from the new

25   container type and thereby at least implicitly designating action methods to be inherited by existing actions from respective parent actions with regard to the new container type in the system; and

storing the new container type including the defined structural behavior and defined specific action methods in a form to be retrieved by the

30   system during run-time.

18. The method of claim 17 wherein storing the new action comprises storing the new container type as a class available to a class factory.

5   19. A method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated including an action method table comprising a plurality of action methods, each 10   action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the method for checking an action during run-time and prior to traversing the directed graph therewith and comprising:

15   determining whether the action must be instantiated;

instantiating the action during run-time if the action is not in fact instantiated;

determining whether the action method table of the action must be expanded and filled in based on any new container types in the system; 20   expanding and filling in the action method table of the action during run-time if the action method table of the action must in fact be expanded and filled; and

employing the action to traverse the directed graph.

25   20. The method of claim 19 wherein the action depends from a parent action, and wherein instantiating the action comprises:

querying a run-time information library for information on the instantiating action, the information including the parent action of the instantiating action and the specific action methods associated with the instantiating action; 30   and

instantiating the action based on such information by:

ensuring that the parent action to the instantiating action is instantiated;

copying the action method table of the parent action and employing the copied action method table as the action method table of the instantiating action; and

updating appropriate ones of the action methods in the copied method table to be the specific action methods associated with the instantiating action based on the information obtained from the run-time information library, wherein at least some of the action methods in the action method table of the instantiating action are not updated and thus are action methods inherited from the parent action.

21. The method of claim 19 wherein the action depends from a parent action, and wherein expanding and filling in the action method table of the action comprises:

determining how many new container types have been added;

expanding the action method table of the action of interest a corresponding number of entries;

querying a run-time information library for information on each added container type and the action of interest, the information including the parent action of the action of interest and a specific action method of each added container type if any with regard to the action of interest; and

filling in the expanded entries of the action method table of the action of interest based on such information by:

ensuring that the parent action to the action of interest is expanded and filled in;

copying the corresponding entries from the action method table of the parent action and employing the copied entries as the corresponding entries in the action method table of the action of interest; and

updating appropriate ones of the action methods in the

5   copied entries in the action method table to be the specific action methods associated with the added container types if any based on the information obtained from the run-time information library, wherein action methods in the copied entries of the action method table of the action of interest that are not updated are action methods inherited from the parent action.

10

       22.   A computer-readable medium having computer-executable instructions thereon for implementing a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on

15   one of a plurality of container types, each action including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the system further having a run-time information library

20   which includes a plurality of information objects, each information object representing one of an action and a container type.

       23.   The medium of claim 22 wherein each information object in the implemented system contains information about the represented action or

25   container type including members of a group consisting of a globally unique identifier, a type name, a system type, a pointer to ancestor type information, a table information pointer for any derived types, a pointer to a class factory which instantiates the type, and combinations thereof.

24. The medium of claim 22 wherein each information object representing an action includes information necessary to instantiate the action at run-time.

5   25. The medium of claim 22 wherein each information object representing a container type includes information necessary to instantiate a container based on the container type at run-time.

26. The medium of claim 22 wherein each information object
10 representing a container type includes information necessary to expand and fill in the action method table of each action at run-time such that a container instantiated based on the container type may be handled when encountered by each action when traversing the directed graph.

15   27. A computer-readable medium having computer-executable instructions thereon for implementing a method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as
20 instantiated including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the method for instantiating an action during run-time, the instantiating action
25 depending from a parent action, the implemented method comprising:

querying a run-time information library for information on the instantiating action, the information including the parent action of the instantiating action and the specific action methods associated with the instantiating action; and

30   instantiating the action based on such information by:

ensuring that the parent action to the instantiating

action is instantiated;

copying the action method table of the parent action

and employing the copied action method table as the action method table of the

5   instantiating action; and

updating appropriate ones of the action methods in the

copied method table to be the specific action methods associated with the

instantiating action based on the information obtained from the run-time

information library, wherein at least some of the action methods in the action

10   method table of the instantiating action are not updated and thus are action

methods inherited from the parent action.

28.   The medium of claim 27 wherein the system further has a

global table including the action methods associated with the instantiated actions,

15   and wherein the action method table of each action comprises a plurality of

pointers pointing to the action methods in the global table, each action when

traversing the directed graph employing the type value of an encountered

container as an offset into the action method table thereof to select the pointer to

the action method in the global table to be executed on the encountered

20   container, the implemented method comprising:

copying the specific action methods associated with

the instantiating action to the global table; and

updating appropriate ones of the pointers in the copied

action method table to point at the specific action methods associated with the

25   instantiating action based on the information obtained from the run-time

information library, wherein at least some of the pointers in the action method

table of the instantiating action are not updated and thus are pointed at action

methods inherited from the parent action.

29.   The medium of claim 27 wherein the system further has a run-time action table maintaining a list of all actions instantiated in the system, the implemented method further comprising first determining from the run-time action table that the action is not instantiated.

5

30.   The medium of claim 27 wherein the actions in the system are organized into an action tree which branches out from a generalized base action to a plurality of more specific actions at multiple levels of specificity, each non-base action directly depending from a parent action based on the action tree, and wherein ensuring that the parent action to the instantiating action is instantiated comprises going back through the action tree from the instantiating action to the base action as far back as is necessary to ensure that the parent action to the instantiating action is instantiated.

15

31.   A computer-readable medium having computer-executable instructions thereon for implementing a method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the method for developing a new action to be instantiated during run-time, the new action for imparting a particular functionality, the implemented method comprising:

defining a parent action for the new action to depend from based on the particular functionality to be imparted;

defining specific action methods to be performed by the new action with regard to at least some of the container types in the system based on

the functionality to be imparted and thereby at least implicitly designating action methods to be inherited by the new action from the parent action, the inherited action methods to be performed by the new action with regard to the other container types in the system; and

5   storing the new action including the defined parent action and defined specific action methods in a form to be retrieved by the system during run-time.

32.   The medium of claim 31 wherein storing the new action
10   comprises storing the new action as a class available to a class factory.

33.   A computer-readable medium having computer-executable instructions thereon for implementing a method in combination with a system having a plurality of actions available for traversing a directed graph, the directed
15   graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table
20   thereof to select the action method to be executed on the encountered container, the implemented method for expanding and filling in the action method table of an action ('action of interest') during run-time upon determining that at least one container type has been added to the system, the action of interest depending from a parent action, the implemented method comprising:

25   determining how many new container types have been added;

          expanding the action method table of the action of interest a corresponding number of entries;

          querying a run-time information library for information on each
30   added container type and the action of interest, the information including the

parent action of the action of interest and a specific action method of each added
container type if any with regard to the action of interest; and

      filling in the expanded entries of the action method table of
the action of interest based on such information by:

            ensuring that the parent action to the action of interest
is expanded and filled in;

            copying the corresponding entries from the action
method table of the parent action and employing the copied entries as the
corresponding entries in the action method table of the action of interest; and

            updating appropriate ones of the action methods in the
copied entries in the action method table to be the specific action methods
associated with the added container types if any based on the information
obtained from the run-time information library, wherein action methods in the
copied entries of the action method table of the action of interest that are not
updated are action methods inherited from the parent action.

34.   The medium of claim 33 wherein the implemented method
further comprises determining that a container type has been added to the
system.

35.   The medium of claim 34 wherein determining that a container
type has been added to the system comprises obtaining a count of all container
types in the system and determining whether the action method table of the action
of interest has a proper size based on the count.

36.   The method of claim 33 wherein the system further has a
global table including the action methods associated with the instantiated actions,
and wherein the action method table of each action comprises a plurality of
pointers pointing to the action methods in the global table, each action when
traversing the directed graph employing the type value of an encountered

container as an offset into the action method table thereof to select the pointer to the action method in the global table to be executed on the encountered container, the implemented method comprising:

5   copying the specific action methods associated with the added container types if any to the global table; and

updating appropriate ones of the pointers in the copied entries of the action method table to point at the specific action methods associated with the added container types based on the information obtained from the run-time information library, wherein at least some of the pointers in the

10   action method table of the action of interest are not updated and thus are pointed at action methods inherited from the parent action.

37.   The medium of claim 33 wherein the actions in the system are organized into an action tree which branches out from a generalized base

15   action to a plurality of more specific actions at multiple levels of specificity, each non-base action directly depending from a parent action based on the action tree, and wherein ensuring that the action method table of the parent action to the action of interest is expanded and filled in comprises going back through the action tree from the action of interest to the base action as far back as is

20   necessary to ensure that the action method table of the parent action to the action of interest is expanded and filled in.

38.   A computer-readable medium having computer-executable instructions thereon for implementing a method in combination with a system

25   having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated including an action method table comprising a plurality of action methods, each action when traversing the directed graph employing the type

30   value of an encountered container as an offset into the action method table

thereof to select the action method to be executed on the encountered container, the method for developing a new container type to be added to the system during run-time, the new container type for imparting a particular structure, the implemented method comprising:

5   defining structural behavior to be imparted based on a container instantiated from the new container type;

defining specific action methods to be performed by at least some existing actions with regard to the container instantiated from the new container type and thereby at least implicitly designating action methods to be

10   inherited by existing actions from respective parent actions with regard to the new container type in the system; and

storing the new container type including the defined structural behavior and defined specific action methods in a form to be retrieved by the system during run-time.

15

39.   The medium of claim 38 wherein storing the new action comprises storing the new container type as a class available to a class factory.

40.   A computer-readable medium having computer-executable

20   instructions thereon for implementing a method in combination with a system having a plurality of actions available for traversing a directed graph, the directed graph including a plurality of containers, each container having a type value and being instantiated based on one of a plurality of container types, each action as instantiated including an action method table comprising a plurality of action

25   methods, each action when traversing the directed graph employing the type value of an encountered container as an offset into the action method table thereof to select the action method to be executed on the encountered container, the implemented method for checking an action during run-time and prior to traversing the directed graph therewith and comprising:

30   determining whether the action must be instantiated;

instantiating the action during run-time if the action is not in fact instantiated;

determining whether the action method table of the action must be expanded and filled in based on any new container types in the system;

5   expanding and filling in the action method table of the action during run-time if the action method table of the action must in fact be expanded and filled; and

employing the action to traverse the directed graph.

10   41.   The medium of claim 40 wherein the action depends from a parent action, and wherein instantiating the action comprises:

querying a run-time information library for information on the instantiating action, the information including the parent action of the instantiating action and the specific action methods associated with the instantiating action;

15   and

instantiating the action based on such information by:

ensuring that the parent action to the instantiating action is instantiated;

copying the action method table of the parent action

20   and employing the copied action method table as the action method table of the instantiating action; and

updating appropriate ones of the action methods in the copied method table to be the specific action methods associated with the instantiating action based on the information obtained from the run-time

25   information library, wherein at least some of the action methods in the action method table of the instantiating action are not updated and thus are action methods inherited from the parent action.

42.   The medium of claim 40 wherein the action depends from a parent action, and wherein expanding and filling in the action method table of the action comprises:

determining how many new container types have been

5   added;

expanding the action method table of the action of interest a corresponding number of entries;

querying a run-time information library for information on each added container type and the action of interest, the information including the

10   parent action of the action of interest and a specific action method of each added container type if any with regard to the action of interest; and

filling in the expanded entries of the action method table of the action of interest based on such information by:

ensuring that the parent action to the action of interest

15   is expanded and filled in;

copying the corresponding entries from the action method table of the parent action and employing the copied entries as the corresponding entries in the action method table of the action of interest; and

updating appropriate ones of the action methods in the

20   copied entries in the action method table to be the specific action methods associated with the added container types if any based on the information obtained from the run-time information library, wherein action methods in the copied entries of the action method table of the action of interest that are not updated are action methods inherited from the parent action.

25